

Міністерство освіти і науки України
Національний технічний університет України «Київський політехнічний інститут
імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи № 7 з дисципліни
«Алгоритми та структури даних-1.
Основи алгоритмізації»

«Дослідження лінійного пошуку в

послідовностях»

Варіант 3

Виконав студент

Баран Софія Володимирівна

(шифр, прізвище, ім'я, по батькові)

Перевірів

(прізвище, ім'я, по батькові)

Лабораторна робота 7

Дослідження лінійного пошуку в послідовностях

Мета – дослідити методи послідовного пошуку у впорядкованих і неупорядкованих послідовностях та набутти практичних навичок їх використання під час складання програмних специфікацій.

Завдання:

3	$55 - 2 * i$	$40 + 3 * i$	Знайти значення, яке є мінімальним та, більшим за задане Р
---	--------------	--------------	--

Постановка задачі: Для виконання поставленої задачі скористаємося підпрограмами для заповнення двох масивів за заданою формулою, для пошуку однакових елементів та заповнення ними третього масиву, виводу масивів, та пошуку мінімального елемента.

Математична модель:

Змінна	Тип	Ім'я	Призначення
Перша змінна	Цілочисельний	size	Початкове дане
Друга змінна	Символьний	arr1	Початкове дане
Третя змінна	Символьний	arr2	Початкове дане
Четверта змінна	Символьний	arr3	Проміжні дані
П'ята змінна	Цілочисельний	min	Результат

Формули, що використовуються:

- $55 - 2 * i$
- $40 + 3 * i$

Псевдокод:

Крок 1: Визначимо основні дії

Крок 2: Деталізуємо умову з використанням підпрограм.

Крок 1:

Початок

size = 10

Заповнення масивів

Пошук min, більшого за P

Кінець

Крок 2:

Підпрограма

fill_arr1_arr2(char arr1[], char arr2[], const int size)

повторити (int i = 0; i < size; ++i)

 arr1[i] = char(55 - 2 * i);

 arr2[i] = char(40 + 3 * i);

все повторити

Кінець підпрограми

Підпрограма

fill_arr3(char arr1[], char arr2[], char arr3[], const int size)

 int index = 0;

повторити

 for (int i = 0; i < size; ++i)

повторити

 for (int j = 0; j < size; ++j)

якщо (arr1[i] == arr2[j])

то arr3[index] = arr1[i];

 index++;

все якщо

все повторити

все повторити

повторити

 for (int i = index; i < size; ++i)

 arr3[i] = '0';

все повторити

Кінець підпрограми

Підпрограма

```
output_array(char arr[], const int size)
    повторити (int i = 0; i < size; i++)
        вивести arr[i]
        вивести " "
    все повторити
```

Кінець підпрограми

Підпрограма

```
minimum(const char arr[], const int size)
    int P, min, elem;
    ввід P;
    min = int(arr[0]);
    повторити (int i = 0; i < size; ++i)
        elem = int(arr[i]);
        якщо (elem < min && elem > P)
            то min = elem;
        все якщо

    якщо (min < P)
        то return 0;
    інакше
        return min;
```

Кінець підпрограми

Початок

```
size = 10;
arr1[size], arr2[size], arr3[size];
fill_arr1_arr2(arr1, arr2, size);
fill_arr3(arr1, arr2, arr3, size);
output_array(arr1, size);
output_array(arr2, size);
output_array(arr3, size);
min = minimum(arr3, size);
вивести min;
```

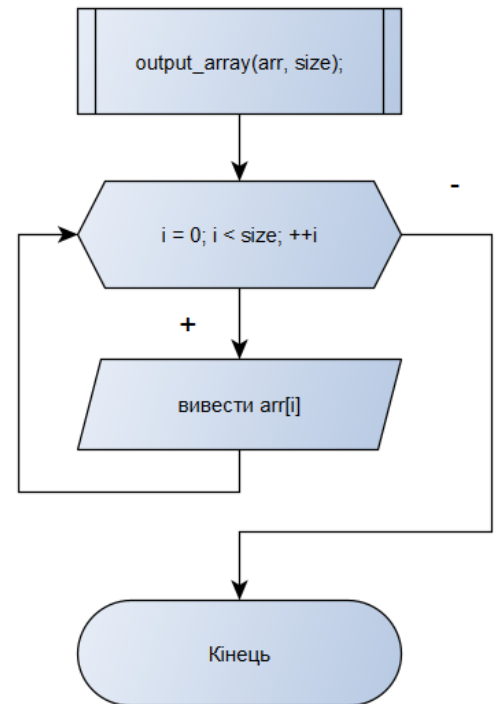
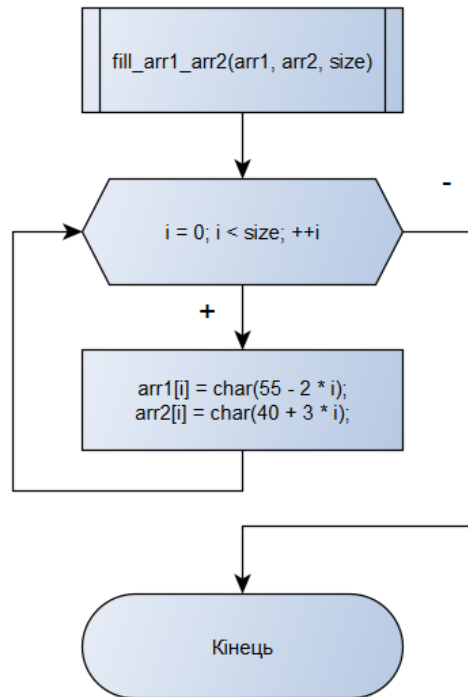
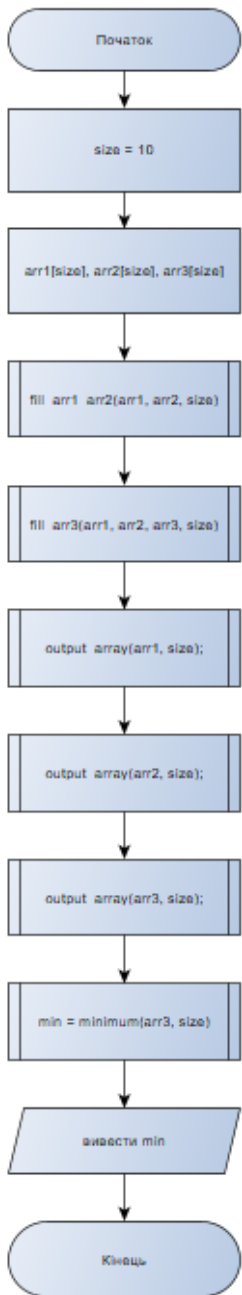
Кінець

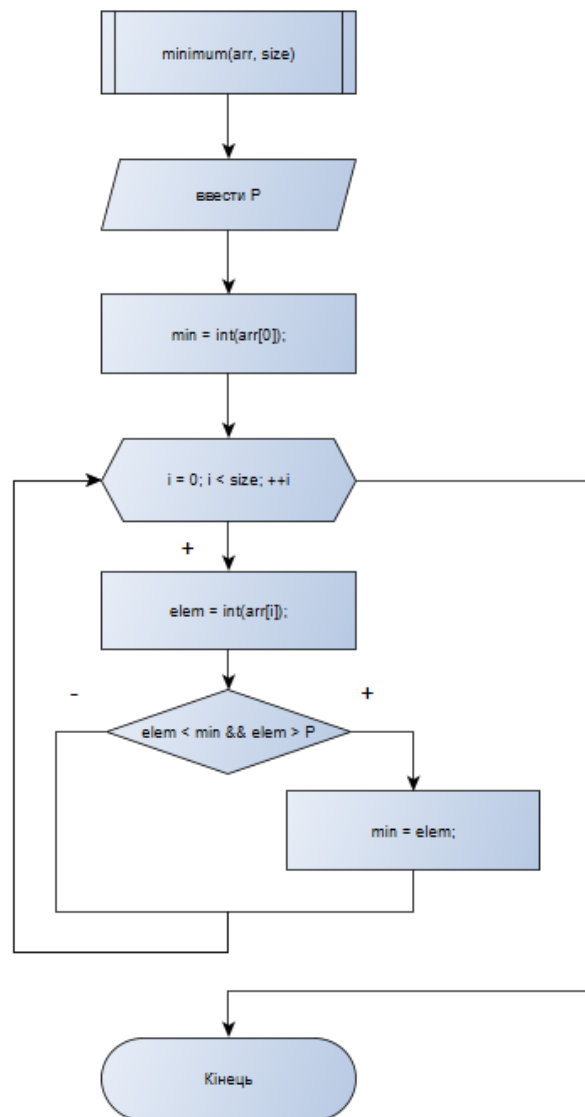
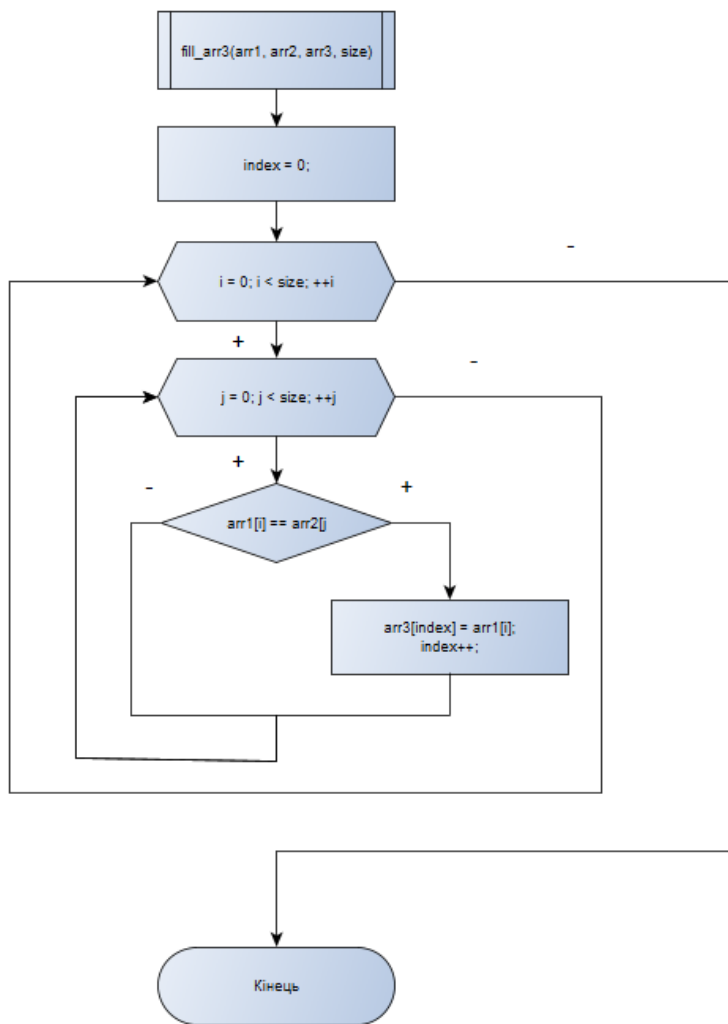
Блок схеми

Крок 1



Крок 2





Код програми на мові C++:

```
#include <iostream>

using namespace std;

void fill_arr1_arr2(char [], char [], int);

void fill_arr3(char [], char [], char[], int);

void output_array(char [], int);

int minimum(const char[], int);

int main() {
    const int size = 10;
    char arr1[size], arr2[size], arr3[size];
    fill_arr1_arr2(arr1, arr2, size);
    fill_arr3(arr1, arr2, arr3, size);
    cout << "First array" << endl;
    output_array(arr1, size);
    cout << "\nSecond array" << endl;
    output_array(arr2, size);
    cout << "\nThird array" << endl;
    output_array(arr3, size);
    cout << "\nMinimum ASCII code that is bigger than P" << endl;
    int min = minimum(arr3, size);
    cout << min << " (" << char(min) << ")";
    return 0;
}

void fill_arr1_arr2(char arr1[], char arr2[], const int size) {
    for (int i = 0; i < size; ++i) {
        arr1[i] = char(55 - 2 * i);
        arr2[i] = char(40 + 3 * i);
    }
}

void fill_arr3(char arr1[], char arr2[], char arr3[], const int size) {
    int index = 0;
    for (int i = 0; i < size; ++i) {
        for (int j = 0; j < size; ++j) {
            if (arr1[i] == arr2[j]) {
                arr3[index] = arr1[i];
                index++;
            }
        }
    }

    for (int i = index; i < size; ++i) {
        arr3[i] = '0';
    }
}

void output_array(char arr[], const int size) {
    cout << "[" << " ";
    for (int i = 0; i < size; i++) {
        cout << arr[i] << " ";
    }
    cout << "]" << endl;
}

int minimum(const char arr[], const int size) {
    int P, min, elem;
    cout << "Enter a P" << endl;
    cin >> P;
}
```



```

min = int(arr[0]);
for (int i = 0; i < size; ++i) {
    elem = int(arr[i]);
    if (elem < min && elem > P)
        min = elem;
}
if (min < P)
    return 0;
return min;
}

```

Виконання програми:

```

First array
[ 7 5 3 1 / - + ) ' % ]

Second array
[ ( + . 1 4 7 : = @ C ]

Third array
[ 7 1 + 0 0 0 0 0 0 0 ]

Minimum ASCII code that is bigger than P
Enter a P
25
43 (+)
Process finished with exit code 0

```

Висновок:

На лабораторній роботі я набув навички роботи з масивами та набув практичних навичок їх використання під час складання алгоритмів лінійного пошуку. У результаті лабораторної роботи було розроблено математичну модель, що відповідає постановці задачі; псевдокод, блок-схеми та код на мові C++, які пояснюють логіку алгоритму.